REMARKS

The Examiner objected to claim 15 stating that to comport to 37 CFR § 1.75, each element or step of the claim should be separated by a line indentation. Claim 15 has been rewritten to cure the defect. Claim 19, which was originally dependent from claim 15 but has been rewritten in independent form, has been rewritten similarly to claim 15.

The Examiner objected to claims 18 and 19, stating that the term "the generator" should be "the generator module". Claims 18 and 19 have been amended to cure the defect.

Claims 5 and 7 have also been reformatted to improve their clarity.

Claims 1-4, 8-11, and 15-18 were rejected under 35 U.S.C. 103(a) as being unpatentable over Fuller, III *et al.* (US 7134081 B2), hereafter "Fuller" in view of Werme *et al.* (US 7171654 B2), hereafter "Werme". Applicant submits that as currently amended, claims 1-4. 8-11, and 15-18 are not obvious in view of the cited prior art.

The Examiner states that Fuller teaches the limitations of claim 1 except for automatically determining a best estimation. The Examiner looks to Werme for the missing teachings. The Examiner maintains that it would have been obvious to modify Fuller with Werme to allow "the functional elements to access the specification information using API calls".

Fuller teaches (Abstract) an instrument I/O assistant and associated algorithms that provide an environment which enables a user to interact with an instrument response without the user having to write parsing code. Fuller also teaches (column 4, lines 57-63) that syntax information from a custom file may be available to help the user to interact appropriately with a specific corresponding instrument. However, Fuller does not teach anything regarding metadata, whether associated with an API call or not. Moreover, Fuller does not teach automatically determining a best estimation of a client grammar communication from an API call, from associated metadata, or from any other data. Furthermore, Fuller does not disclose any ambiguity or inadequacy in the specification of client program grammar by an API call.

Werme teaches (Abstract) a method for distributing information between a plurality of hosts, including the step of preparing specification files "in a language providing a syntax adapted to describe application, system and network specification information". However, Werme does not mention metadata, whether associated with an API call or not, does not teach automatically determining a best estimation of a client grammar communication from an API call, from associated metadata, or from any other data, and does not disclose any ambiguity or inadequacy in the specification of client program grammar by an API call.

Applicant submits that the full extent of the limitation which the Examiner admits is missing from Fuller is the requirement that **when** metadata is associated with the API call, a best estimation **of the client program grammar communication** is automatically determined **from the associated metadata** and from the API call, and that when such metadata is not associated with the API call, a best estimate of the client program grammar communication is obtained from the API call. The Examiner points to col.11, line 64-col.14, line 54 of Werme for the missing teachings, drawing attention to the discussion of instrumentation APTs and API calls.

First, claim 1 has been amended to make it clear that the API call does not fully and unambiguously specify the client program grammar. This limitation is not taught in Fuller or Werme, which assume that the API call is sufficient to determine the grammar. A system in which an API call fully and unambiguously specifies the client program grammar, does not require a **best estimation** of the client program grammar, and hence, there would be no need to include such a mechanism in these prior art systems.

Second, there is no mention in the cited paragraph in Werme of any metadata (meaning data about data) let alone of metadata associated with an API call, which determines how the best estimation may be made. Claim 1 requires that determination to be based on whether or not metadata is associated with the API call. In this regard, it should be noted that Fuller is also silent regarding any metadata associated with any API call. In order to further clarify the significance of metadata to the present invention, claim 1 has been amended to make explicit the step of determining whether or not metadata is associated with the API call.

Third, there is no teaching in the cited passage in Werme of the estimation of any client program grammar communication, from either metadata or the API call itself. The only mention of "grammar" occurs in the passage in column 14, lines 36-43, which teaches that instrumentation correlators FG26A-N "provide grammar-driven capabilities for correlating, combining, and reformatting application data into higher-level metrics (composite events) for use by displays or other components of the Resource Management Architecture. Each Correlator reads in a user-specified correlation grammar file which is interpreted at run-time by the Correlator's instrumentation correlation engine", and hence, the grammar is known. At most, this passage teaches that the grammar in which the application data are received determines the details of how that data are correlated, combined and reformatted, but it does not teach the estimation recited in the claim.

Fourth, there is no teaching in the cited passage in Werme regarding any **automatic** determination or estimation.

Fifth, Applicant submits that the advantage that the Examiner suggests would be achieved – allowing "the functional elements" to access network specification information using API calls – would be irrelevant to the system of Fuller. The "specification information" in Werme is created by the user and consists of 1) application software system structure, capabilities, dependencies, and requirements; and 2) hardware system (computer and network) structure, capabilities, and configuration. Fuller teaches a method of controlling an instrumentation system with simplified interfaces between the user and the instruments. The Examiner has not pointed to any teaching that any "functional elements" in the system of Fuller would benefit by accessing network specification information, let alone that the best way of providing such access would be via API calls.

Claim 2 depends from claim 1 and further requires that the API call is a .NET API call, The Examiner points to col. 24, lines 53-60 of Fuller as providing this teaching. The cited passage concerns two National Instruments software packages that include tools for monitoring API calls. Applicant maintains that the Examiner has not pointed to any teaching that any API call that is monitored is a .NET API call. Hence, there are additional grounds for allowing claim 2.

Claim 4 depends from claim 1 through claim 3, and further requires evaluating the obtained best estimation of the SCPI communication for conformance of the best estimation of the SCPI communication to SCPI specifications. The Examiner points to Werme (col.11, line 64-col.14, line 54) as providing this teaching, drawing attention to the discussion of instrumentation APTs and API calls.

At most, the cited passage teaches than an Instrumentation Daemon reformats data "into the standard internal Instrumentation message format" which "includes a header, a format string describing the application-provided data contained in the message, and the actual data values" but it does not teach that this format is equivalent to the Standard Commands for Programmable Instrumentation (SCPI) protocol standard recited in the claim. Moreover, even if the format taught by Werme were presumed to be identical to SCPI, reformatting data into SCPI format is not equivalent to evaluating an API call for a best estimation of the SCPI communication to SCPI specifications, as the claim requires. Hence, Applicant submits that there are additional grounds for allowing claim 4 and the claims dependent therefrom.

As to claims 8-11, which the Examiner states are computer readable memory device claims corresponding to the method claims 1-4, it should be noted that claim 8 has been amended similarly to claim 1 to specify determining whether or not metadata is associated with the API call, and to specify that the API call does not fully and unambiguously specify the client program grammar. Applicant submits that, as discussed above with respect to claim 1, the combination of Fuller and Werme fails to teach these limitations, as well as failing to teach the requirement that when metadata is associated with the API call, a best estimation of the client program grammar communication is automatically determined from the associated metadata and from the API call; and when there is no metadata associated with the API call, a best estimation of the client program grammar communication is automatically obtained from the API call.

Hence, Applicant submits that these claims are not obvious . Moreover, as discussed above with respect to claims 2 and 4, Applicant submits that there are additional grounds for allowing dependent claims 9 and 11.

As to claims 15-18, which the Examiner states are system claims corresponding to method claims 1-4, it should be noted that claim 15 has been amended similarly to claim 1 to specify determining whether or not metadata is associated with the API call, and to specify that the API call does not fully and unambiguously specify the client program grammar. Applicant submits that, as discussed above with respect to claim 1, the combination of Fuller and Werme fails to teach these limitations, as well as failing to teach the requirement that when metadata is associated with the API call, a best estimation of the client program grammar communication is automatically determined from the associated metadata and from the API call; and when there is no metadata associated with the API call, a best estimation of the client program grammar communication is automatically obtained from the API call.

Hence, Applicant submits that these claims are not obvious. Moreover, as discussed above with respect to claims 2 and 4, Applicant submits that there are additional grounds for allowing dependent claims 16 and 18.

Applicant notes that claims 5, 7, 12, and 14 are allowed over the prior art of record, subject to a final search.

The Examiner stated that claims 6, 13, 19, and 20 would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims subject to the objections detailed above, and subject to a final search. The claims have been rewritten accordingly.

Respectfully Submitted,

Calvin B. Ward
Registration No. 30,896
Date: December 22, 2008

Agilent Technologies, Inc.
Legal Department, M/S DL429
Intellectual Property Administration
P.O. Box 7599
Loveland, CO 80537-0599
Telephone (925) 855-0413
Telefax (925) 855-9214